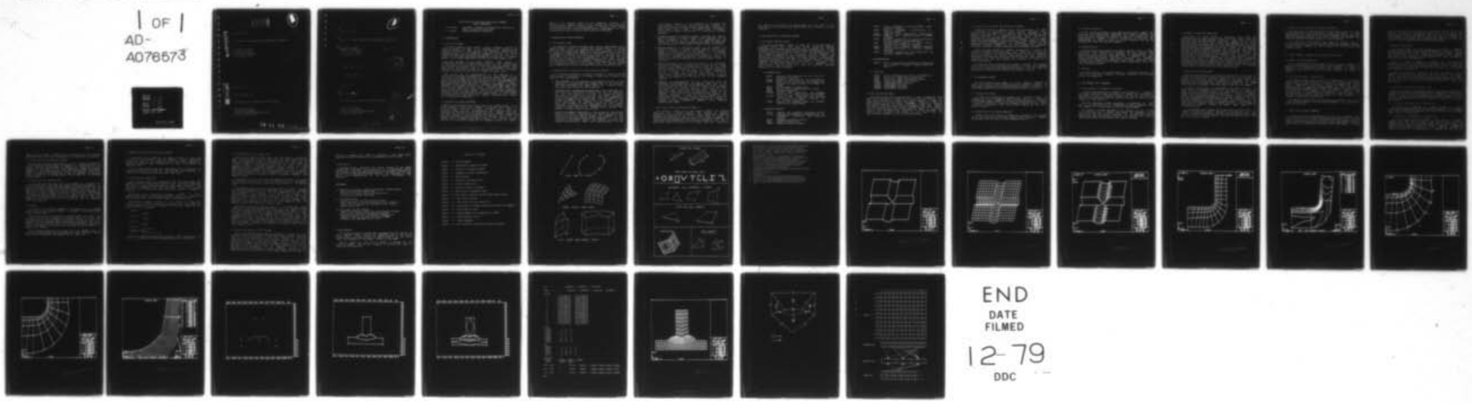
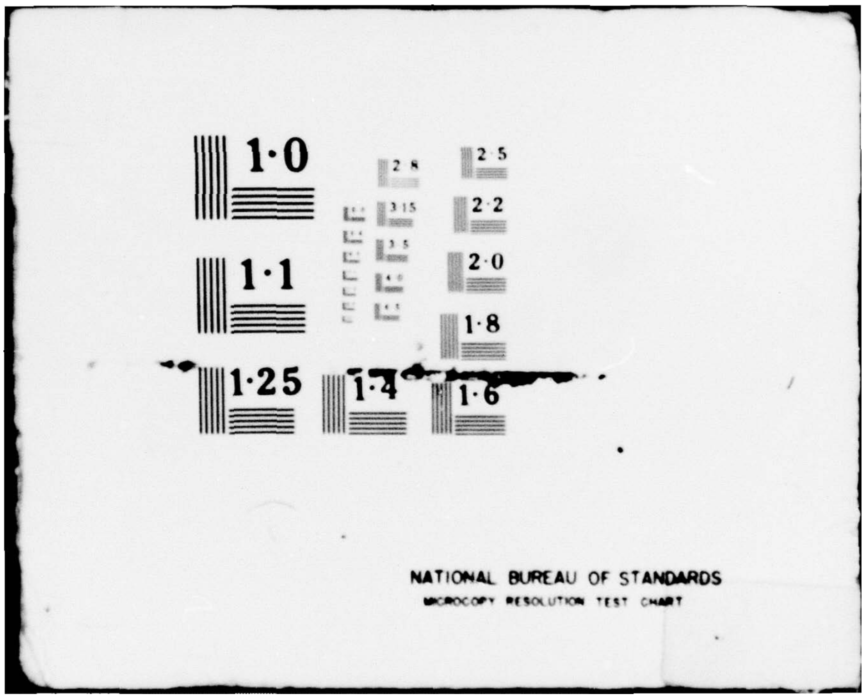
AD-A076 573 ARIZONA UNIV TUCSON DEPT OF AEROSPACE AND MECHANICA--ETC F/G 13/13
DIGITIZING FOR COMPUTER-AIDED FINITE ELEMENT MODEL GENERATION.(U)
OCT 79 H A KAMEL , Z NAVABI N00014-75-C-0837
UNCLASSIFIED TR-5 NL

1 OF 1
AD-
A076573



END
DATE
FILMED
12-79
DDC


1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

LEVEL

N00014-75-C-0837

DIGITIZING FOR COMPUTER-AIDED FINITE ELEMENT MODEL GENERATION

H.A. Kamel & Z. Navabi
University of Arizona
Aerospace and Mechanical
Engineering Department
Tucson, Arizona 85721

October 10, 1979

Technical Report No. 5

Approved for public release, distribution unlimited

Department of the Navy
Office of Naval Research
Structural Mechanics Program (Code 474)
Arlington, Virginia 22217

15)

N00014-75-C-0837

6)

DIGITIZING FOR COMPUTER-AIDED FINITE ELEMENT MODEL GENERATION.

10)

H.A. Kamel & Z. Navabi
University of Arizona
Aerospace and Mechanical
Engineering Department
Tucson, Arizona 85721

12) 35

11) 10 Oct 79

14) TR-5

October 10, 1979

9)

Technical Report, No. 5

Approved for public release, distribution unlimited

Department of the Navy
Office of Naval Research
Structural Mechanics Program (Code 474)
Arlington, Virginia 22217

402 192

DIGITIZING FOR COMPUTER-AIDED FINITE ELEMENT MODEL GENERATION

H.A. Kamel, Professor, Aerospace and Mechanical Engineering Dept., University of Arizona,
Z. Navabi, Graduate Research Associate.

1. INTRODUCTION

1.1 Outline of Paper

The term "computer aided design" conveys different meaning to different people. To some it is synonymous with computer-aided drafting. To others it may describe the geometric definition of complex objects, or the numerical analysis required for design validation. One cannot escape being reminded of the old tale of a group of blindfolded individuals attempting to describe an elephant based on their manual exploration of what the object seems to be.

For the purpose of this paper computer aided design is acknowledged as being a complex process involving many disciplines. In some design activities certain disciplines play a greater role than others, and therefore tend to be overemphas~~ised.~~ This paper describes an approach to a specific discipline, that of structural analysis, which has evolved with the design process in mind.

The vehicle for the implementation of this approach is a collection of programs developed over the last eight years at the University of Arizona, and called the GIFTS system. ~~(1).~~ The programs are in the public domain and are distributed via a users ~~group.~~ This paper briefly describes the latest version of the system, GIFTS-5, and demonstrates its suitability in a design environement by simple examples. The programs constituting the GIFTS system ~~have been~~ were used as a tool for research in many areas, including mesh generation, finite element data base design, interactive analysis, numerical analysis, as well as digitization. The last sections ~~of this paper~~ describe a newly developed digitizing program aimed at facilitating the preparation of input describing complex two-dimensional models for the GIFTS mesh generator.

1.2 Analysis as a Part of Design

Modern analysis methods, of which the finite element method is a prime example, permit the engineer to analyze most types of structures under a variety of loads. Although this is valid in principle, many practical considerations are involved which may influence the desirability and practicality of a computer analysis. The time required to set up the model, the time and cost needed to perform the analysis, the availability of analysis results to the design team at the appropriate time to help in the decision making process, and the general question of adequate return for effort

spent in the analysis phase are all important factors. The availability of computer graphics, time sharing and minicomputers, and new hardware such as array processors, is having a great impact on the design scene. In the following section requirements for the successful marriage of analysis and design are discussed.

2. ANALYSIS AS A PART OF DESIGN

2.1 The Overall View

The design of an object proceeds from a set of requirements to a simple concept, which is successively refined until it reaches maturity. Once a particular aspect of the design is fixed, so as to be linked to a design requirement which may be numerically verified, it becomes advantageous to perform an analysis to check on the requirement. It is clear that the analysis has to develop hand in hand with the design, so that the mathematical model used for the analysis may be refined to correspond to progress in object design. If alternative concepts are considered, the analysis has to provide the corresponding results to help compare them. An analysis system which supports such an activity must have certain characteristics.

2.2 Requirements for Design-oriented Analysis Software

For an analysis system to be useful in design, it should satisfy certain general requirements. An attempt is made here to isolate and describe such requirements.

a. FAST RESPONSE. For the analysis to be effective it has to be responsive, and predictably so. Answers have to be provided when they are needed. The speed should be a factor which encourages experimentation and innovation.

b. INTERFACE WITH A MASTER DESIGN DATA BASE. This is perhaps the most difficult requirement to satisfy, and yet an important one. The time wasted by the user in recreating object descriptions, already part of another data base, represents one of the obstacles to analysis. The lack of communication between the members of a design team is detrimental to the overall objectives. The reasons for the difficulties are many. The problem of data base definition and proper management is substantial. The management of the extremely large data bases, required for a complex design, places high demands on today's state of the art [2].

c. ANALYSIS REFINEMENT CAPABILITY. In the process of preliminary design the analysis requirements are quite different than those during the final design stages. For example, if two alternative configurations are being considered, it is only necessary to use simple models, the results of which are compared qualitatively rather than quantitatively. In the

final stages, however, it is necessary to introduce all relevant detail, and expect sufficient accuracy to assess the characteristics of the final object adequately. Upgrading a model means many things, including mesh refinement, introduction of detail, and the use of alternative analysis software.

d. ITERATIVE NATURE. An analysis package, suitable for design, must be iterative in nature. A model being analyzed has to remain accessible to the user for modifications and reanalysis. In this manner continuous refinement is possible, and the evaluation of the effect of modification is feasible.

e. MULTIPLE PATH. At a particular point of the analysis it should be possible to choose between several options. For example, a static load may be applied, or the dynamic properties of the model may be evaluated. If a structural optimization is underway, the user may choose to run a few cycles of a standard mathematical optimization method, or intefere manually by modifying the structural model based on intuition.

f. GRAPHICS ORIENTED. For the analysis process to be effective, the natural language of graphics has to be used to the utmost possible extent. Numerical output should only be restricted to few significant variables (e.g. the maximum deflection, and its location). To understand the load transmission pattern, or judge when a weight minimization is to be interrupted, a plot of the model or weight versus iteration number is far superior to a full print out of all analysis results.

g. OPERABLE ON MINICOMPUTERS. For the interaction with the analysis to be satisfactory, the response times have to be both sufficiently fast and fairly predictable. In the opinion of the author, the current trend of acquisition of small dedicated, well configured systems will continue. The important reason here is that an interactive system has to operate at a low load factor, purposely wasting some of the computer resources in order to provide satisfactory and predictable response. In addition, devices such as array processors will complement stand-alone design systems to produce relatively inexpensive computers with adequate computational power.

2.3 Who Should Use the Analysis Software ?.

The question arises as to who should conduct the analysis associated with a design endeavor. The process of analysis automation by such devices as automatic mesh generation, graphics, mathematical optimization, and so on, often leads to misuse. An analysis program should always be placed in the hands of someone who understands the fundamentals underlying the program in use. Such a person may indeed be a designer with a sound analytical background,

or a specialist attached to the design team. In this case, as in many others, the new technology imposes additional demands on user training.

3. THE STRUCTURE OF THE GIFTS PACKAGE

3.1 The GIFTS Program Library

The GIFTS programming system is made out of a multitude of programs, which fall into four broad categories namely: preprocessing, analysis, postprocessing and utilities. Any of the programs, also called modules, may be executed independently. Checks at the beginning of each module ensure that the prerequisites for the requested computation are satisfied. Certain sequences of modules are recommended in order to perform particular tasks. Some of the modules are batch programs, and others operate interactively. Graphical representation of data is provided whenever possible, although alphanumeric playback of stored data is available. Error checking is performed extensively, and error diagnostics are user-oriented. The following is a list of currently available modules in the GIFTS-5 program:

A. PREPROCESSING

BULKM	Automatic mesh generator for two- and three-dimensional surfaces.
BULKS	Automatic mesh generator for solid models.
EDITM	Model editor for two- and three-dimensional surfaces, and model builder for trusses and frames.
DEFCS	Defines model as substructure.
BULKF	Assigns global freedom pattern to nodes.
OPTIM	Optimizes bandwidth.
BULKLB	Defines loads, tempratures and boundary conditions for models generated with BULKM.
EDITLB	Defines loads, tempratures and boundary conditions for all models. Displays load and boundary conditions.
LOADS	Defines and displays loads and boundary conditions for solid model.

B. ANALYSIS MODULES

STIFF	Computes and assembles stiffness matrix.
STIFFX	Computes and assembles stiffness matrix for axisymmetric structures under arbitrary loading.
DECOM	Decomposes stiffness matrix.
DEFL	Computes deflections.
STRESS	Computes element stresses.

SOLAX	Solves axisymmetric solids problems, under arbitrary loading.
REDCS	Reduces (condenses) a substructure, its loads and masses.
LOCAL	Computes local substructure deflections.
RESIDU	Computes force residues on nodes to assess accuracy.
AUTOL	Automatic generation of initial load vectors for the subspace iteration method.
SUBS	Performs one iteration to obtain natural frequencies and modes of vibration.
TRAN1	Prepares for transient response analysis.
TRAN2	Continues preparation for transient response analysis.
TRANS	Performs and monitors time integration for transient response analysis.

C. POSTPROCESSING

RESULT	Plots and provides information on deflection and stresses in trusses, frames, plates and shells.

D. UTILITIES

SAVEK	Saves stiffness matrix before decomposition.
DUMPM	Selectively dumps model information.
DUMPI	Selectively dumps intermediate results.
DUMPR	Selectively dumps final results.
PIGSAP	GIFTS/SAP4/GIFTS interface.
GFTSTA	GIFTS/STAGS interface.
GFTANS	GIFTS/ANSYS interface.
GFTMOV	GIFTS/MOVIE interface.

3.2 The Unified Data Base

The GIFTS program stores problem data in a user transparent data base, which consists of a number of files. The contents of each file are clearly defined for a particular GIFTS version. A number of GIFTS library I/O routines help a programmer, involved in writing new system modules, to extract information from, and deposit information to the data base. The well documented data base, coupled with the availability of the library routines, makes it relatively easy to create new modules and interfaces to other programs. The ordinary user, however, has access to the data only via user-oriented commands to one of the GIFTS modules.

3.3 Interfacing to Linear and Nonlinear Programs

As mentioned above, it appears fairly straightforward, at least in principle, to interface to other analysis programs. This satisfies the requirement that analysis may be upgraded as the design progresses. It is possible to make use of specialized programs, such as nonlinear codes, optimization codes, and so on. Interfaces are already available to many programs, such as SAP4, STAGS, and MOVIE-ARIZONA, a continuous tone color graphics hidden line and surface display package. (A minicomputer version of the MOVIE-BYU program.) Members of the users group have produced in-house interfaces to other programs such as NASTRAN and ADINA.

There is a basic difference between interfacing to linear and nonlinear programs. In the nonlinear case, where the solution is carried in steps, and a certain amount of monitoring is necessary during the analysis, the restart requirements are more rigorous. Furthermore the analysis code itself has to provide some means of interaction with the user during the solution, at least in a restricted fashion.

The difficulties of maintaining a program interface are greater than those for a single program, particularly if the two programs are in the process of being updated and modified. This fact must be considered in designing and maintaining a program library.

4. THE ELEMENT LIBRARY

The GIFTS package has a number of finite elements designed to enable the analysis of most standard structural models. A summary of the elements supported in the analysis mode is shown in figure 1. A brief description of the structural models handled is given here.

4.1 Trusses and Frames

Two types of one-dimensional elements are available within the program. The axial rod members, ROD2 and ROD3, and the 6 degree of freedom beam element, BEAM2. To define the beam properties, a number of standard cross-sections may be addressed. They include an I-beam, a channel, a Z-section, right and oblique angles as well as a general polygon.

4.2 Membranes and Stiffened Membranes

Constant and linear strain triangles are included in the system, as well as first and second order quadrilateral membrane elements. Rod elements may be used to stiffen membrane models.

4.3 Plates and Shells

A flat triangular element, and a quadrilateral plate element are available. Both elements have membrane and bending stiffness. To avoid the usual difficulties associated with the absence of the sixth degree of freedom, artificial stiffening is provided. The elements may be used for flat or curved surfaces and may be stiffened by using rod and beam elements.

4.4 Substructuring

Substructuring is provided in the static analysis case. A model may be built from ordinary finite elements and more complex ones describing large regions, and specified as substructures. The detailed substructures are generated under a separate job name. Cross referencing between data bases provides the necessary information to assemble the condensed substructure properties onto the main model, and extract the substructure deflections later for local analysis purposes.

4.5 Solids

For solid models, only preprocessing is currently available. In that, only first order elements (4-noded tetrahedron and 8-noded bricks) are supported.

5. THE GENERATION OF GEOMETRY

5.1 The Concept of a Geometric Hierarchy

In GIFTS, mesh generation is based on the concept of a geometric hierarchy. First key nodes are defined by number and coordinates. Key nodes are used to generate curves. The curves are used to generate surface patches, and the patches are used to define solid chunks [3]. (See figure 2.)

The logic employed in model generation is preserved for later use, as in load definition. It is possible to apply a pressure load, for example, on a surface patch using one single command.

The geometric concepts employed in the program are well suited for an interface with a design data base. The parametric curve description, and the key point locations can be extracted from a master data base. Decisions relating to grid density and relative node spacing need only be made at the time of analysis.

5.2 Computer Aided Mesh Generation

Key points are first defined or digitized. Then curves are generated so as to pass through a given set of key nodes. The key nodes are only used for geometric reference, except for the first and last nodes of each curve, which are assumed active. Curve generation and discretization, for the purpose of analysis, are done simultaneousely. To discretize the curve, a number of intermediate points are generated automatically on each curve by the program. They may be equally spaced or biased. The discretized form of the boundary lines are used to generate surface patches, directly in discretized form. The shape of the boundary curves of a grid, and the spacing of the boundary points, controls the geometry of the discrete patch and the distribution of the automatically generated internal grid nodes. In solid mesh generation, the process is extended through one more dimension. The internal subdivision of a volume bound by several surface patches is performed by creating one two-dimensional layer of points after another, from the boundary curves, which are extracted from the surface patches.

5.3 Model Editing and Display

Whereas mesh generation is suitable for structures which may be broken into topologically regular arrays of points and elements, a process of editing and detailed generation of single nodes and elements, or strings thereof, is necessary to complement the large scale generation process. This capability is provided by the EDITM program. Both the keyboard and cursor may be used for the purpose.

In addition to editing, it is necessary to be able to plot the model overall configuration, as well as any amount of detail, in order to verify the model before the analysis may be conducted. EDITM allows the plotting of patch boundaries, or detailed element shapes. The view may be rotated to any desired angle, prespective may be introduced,and two-dimensional close-ups (windowing) as well as three-dimensional close-ups (boxing) may be performed. One may activate or deactivate a patch or an element type. It is possible to label the points by their user or system (internal, after bandwidth optimization) numbers, elements by number, type, material or thickness values.

In solid models, the display of all model details would result in crowded undecipherable plots. Therefore only selected point and element arrays are displayed at any one time together with the chunk outlines. Labeling is only applied to points and elements which form part of the active display.

5.4 Application of Loads and Boundary Conditions

The application of load and boundary conditions follows the same approach employed in mesh generation. Global application of loads and boundary conditions may be performed by BULKLB, whereas detailed loads and boundary conditions may be introduced via EDITLB. The LOADS program performs both functions for solid models.

It is possible, in both EDITLB and LOADS to display loads in arrowed vector form, and active degrees of freedom as simple dashes. In solid models, display is only possible on active point and element slices.

6. Solved Example: Computer-aided Structural Analysis

6.1 Model and Load Generation

The three-dimensional model shown in figure 3, which consists of eight four-sided grids, was generated by the commands of figure 4. Details of the generated elements are displayed in figure 5. Pressure loads, applied to two of the eight surfaces, are shown in figure 6.

6.2 Structural Design, a Case Study

The simple bracket shown in figure 7 is subjected to a horizontal shear load at the top, and built-in at the left bottom end. Stress contour results, shown in figure 8, indicate von-Mises stress values equal to 280% of a given datum stress. To reduce the stress concentration, the shape of the bracket midsection is redefined by relocating some boundary nodes using the terminal cursor, figure 9,to produce a new geometric configuration. In addition, the thickness near the inner fillet was increased, as seen in figure 10. The new stress values, as shown by figure 11, have dropped to a maximum of 190% of the datum stress.

The complete process from initial generation through three design cycles was performed on a time-shared minicomputer in a period of less than two hours.

7. DIGITIZING OF BASIC GEOMETRY

In observing typical mesh generation activities, it becomes clear that the bottleneck is in specifying key point coordinates, and the logical structure of the mesh. Furthermore, much of the mesh density input contains unnecessary duplication. A new experimental program, called DIGIT, was developed to overcome these shortcomings. It uses

a digitizing tablet to enter key point positions, specify curves and patches, and uses the grid generaion logic in order to minimize the input of grid density parameters. Currently it only applies to two-dimensional structures, but the basic logic may be extended to three-dimensional problems. The program may be changed so that the terminal cursor substitutes for the digitizing tablet.

7.1 General Description

The program DIGIT has been designed to digitize a complex two-dimensional model, provide graphical feedback to the user of all actions performed, and deposit the final logic in a standard command file suitable for use by the BULKM mesh generator. This section describes the user perception of the program, with some explanation of the basic algorithms employed.

First, the user is prompted to perform an initialization procedure, in which three points are selected from the plot and their locations are specified numerically. This initial information thus provided is used to set the scale and orientation of the model with respect to the tablet.

Once the initialization procedure has been performed, a menu appears on the screen, see Figure 12. The user may at this point select from many different program functions by pointing the stylus of the tablet at the location corresponding to the appropriate menu box. The available functions, and a brief description of each, are given below.

7.1.1 Key Point Generation (Function KP)

Once in this mode, the user may digitize any number of key points by moving the stylus to the position of the node, and pressing it against the tablet surface. The physical coordinates of the point are then entered into the data base, a small square appears on the screen at the appropriate location, and a bell sounds to reinforce the feedback.

7.1.2 Straight Line Generation (Function SL)

Circular arcs are generated in a similar manner to straight lines, except that three points are picked instead of two. Graphic and acoustic feedback is provided, and error checks are performed before the arc is finalized.

7.1.3 Composite Line Generation. (Function CL)

One of the useful features of GIFTS is the ability to connect several lines and arcs to form a new logical unit, called a composite line. Such a unit is then treated as a single curve, and may specify the edge of a patch. In order to specify a composite line, the user picks the end points of the component segments in sequence The last one is picked twice to signify the end of the

composite line. Again, a composite line is repainted in its entirety upon completion, and a bell is sounded. Error detection is provided.

7.1.4 Grid Generation Functions (G4,G3)

By entering the four corner points of four already established lines, forming a closed polygon, in sequence, a four-sided grid is defined. Switches can be preset to specify the properties of the elements to be used in the mesh generation. Upon successful completion a dashed quadrilateral, joining the midpoints of the four sides, is drawn on the screen, and a bell is sounded. Three-sided grids are generated in the same manner. Error detection is provided in both cases.

7.1.5 Choice of Grid Density (DV)

In this mode the user may enter the number of divisions for selected grid sides. The sides are selected by program prompting, in which the program redraws the line segment and waits for the user to specify the number of partitions for the segment specified. If the user does not wish to specify a certain line density, he requests an alternative line by a push on the stylus. The program continuously circulates through all edges until all density values have been specified. The program requires only a minimum of data to find the number of divisions for all lines. The algorithm used is described in the next section.

7.2 Example

In this section we show an example of the generation of a BULKM input file, by the program DIGIT, to produce a two-dimensional finite element model.

First, the key point locations are entered using the tablet and stylus, see Figure 12. A number of straight lines and circular arcs are specified, as described above, to produce figure 13. Note the presence of three composite patch sides. Figure 14 shows the final screen display, after the grids have been successfully specified. Finally, the number of divisions for only seven of the lines were to specify grid density.

After program termination, the input file for BULKM, shown in figure 15 was generated. Figure 16 shows the mesh generator output as a result of executing the command file of figure 15.

8. COMPUTER AIDED SELECTION OF GRID DENSITY

To determine the grid density, the number of nodes on each side of the grid (side density) must be defined. Certain constraints control the choices, as described below. The DIGIT program is designed to exploit these constraints in order to minimize the required input.

A set of simultaneous equations, representing the constraints, is constructed and used to solve for the unknown line densities in terms for the known ones, and additional user input.

8.1 The Constraint Equations

Three different types of constraints may be encountered. Two opposite sides of a four-sided grid must have identical densities. The sides of a three-sided grid must all have equal densities. The density of a composite line must be equal to the sum of the densities of the constituent lines, minus the number of segments, plus one.

The above three constraints introduce a number of linear dependencies among the line densities, which reduce the amount of required user input. The following example demonstrates this.

A structure is shown in figure 17. The letters on each line indicate the line density. There are four grids, ten line segments, and one composite line in this example. The following equations can be written for the problem:

Grid # 1 :

$$d = h$$
$$h = f$$

Grid # 2 :

$$f = g$$
$$i = c$$

Grid # 3 :

$$g = e$$
$$e = j$$

Grid # 4 :

$$a = b = k$$

Composite line :

$$k = h + i + j - 2$$

Since nine equations are present, only two input parameters are needed to determine all eleven unknown densities "a" through "k".

8.2 Interactive Solution Algorithm.

From the line and grid definitions, an n by n table is generated, where n is the number of lines. (see figure 18.) Each column of the table corresponds to a line. If the density of a line, say line L1, is equal to that of another line, L2, an entry of 1 is created in row L2 of column L1 (i.e. position [L2,L1] of the table). If line segments L1,L2,L3,... form the composite line CL, entries of 2 are generated in table positions [L1,CL], [L2,CL], [L3,CL],... etc.. Next the table is replaced, after some manipulations, by a pointer list, and a composite line list. In addition, a list is created for the purpose of storing the user provided, or program computed, line density values.

In the pointer list, each entry corresponds to one of the lines. The contents of the list are pointers referencing the appropriate density values for the given lines, which are stored in a density list.

The composite line list, also generated from the table, contains two rows. Each composite line is represented by a number of columns, each corresponding to one of the line segments, and the last corresponding to the composite line itself. Zero columns are inserted between composite lines, and act as separators. The first entry in each column contains a pointer to the density value associated with the line in question. To distinguish between the pointers of the segments and the pointers of the composite line, the segment pointers are assigned a negative value.

.During the interactive specification of line densities, these lists are continuously updated and checked out. The program scans the density value list to find a zero (non-specified) value. If one is found, the user is given the option to assign a value to it. When a value is entered for the density of a particular line, the corresponding entry in the density list is filled. Next the composite line list is scanned, and if the value is referenced, an entry of 1 is placed in the second row of the composite list. Each time a new value is entered, the composite line list is searched. If only one unknown is present, the unknown is calcualted and entered in the density list.

8.3 Further Automation of the Process

The process described above may be extended to relieve the user from more detail. For example, the user may assign certain weighting factors to various key points in the grid, denoting the relative importance of the surrounding areas. A target number for the grid nodes may be given for the entire model, and the program then proceeds to automatically assign line density values to all lines. The ability to bias grids within the GIFTS system may also be addressed. Bias parameters may be assigned to the lines automatically by the program. A merit function, related to the importance factors, and the resulting element shapes, may be used to

provide an optimum grid aimed at providing a fine mesh where required, without distorting the grids beyond a certain limit.

9. Conclusion

The paper briefly describes the latest version of the GIFTS program, and discusses its suitability in a preliminary design environement. A new two-dimensional digitizing program, intended for use with an already existing mesh generator, is described in some detail, and recommendations for its further development are outlined.

REFERENCES

[1] GIFTS-5 Users Manual, Theoretical Manual, Systems Manual, Modelling Guide and GIFTS Primer, Interactive Graphics Engineering Laboratory, University of aizona, July 1979.

[2] R.R. Heldenfels, Automation of the Aircraft Design Process, Paper presented at the XIth Congress of the Industrial Council of the Aeronautical Sciences, Haifa, Israel, August 25-30, 1974.

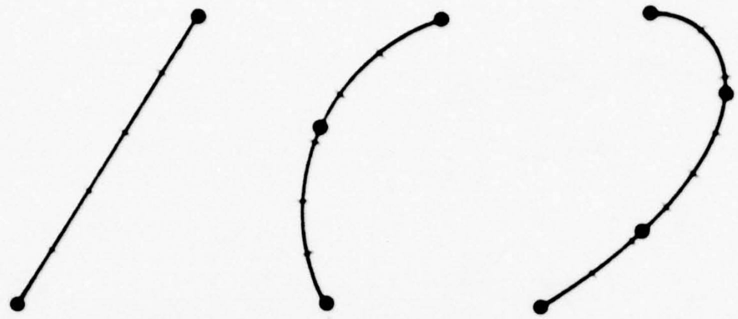
[3] H.A. Kamel and M.W. McCabe, Geometry and Function Definition for Discrete Analysis and its Relationship to the Design Data Base, Presented at the first International Symposium on Computer-Aided Hull-Surface Definition, SNAME, Annapolis, Md.,September 26-27, 1977.

Acknowledgement

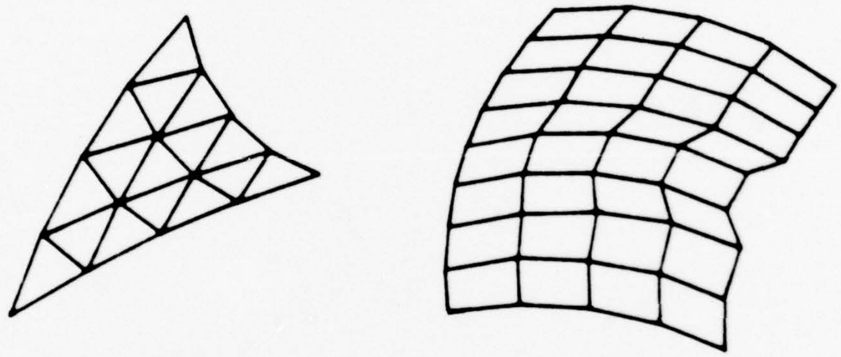
The authors gratefully acknowledge the support of the Office of Naval Research under contract no. N00014-75-C-0837, the United States Coast Guard under contract no. DOT-CG-43565-A and the Gifts Users Group members for the support leading to this paper.

Special thanks are also due to Michael W. McCabe for his technical support, and to Ms. Cynthia B. Francis for help in preparing the manuscript.

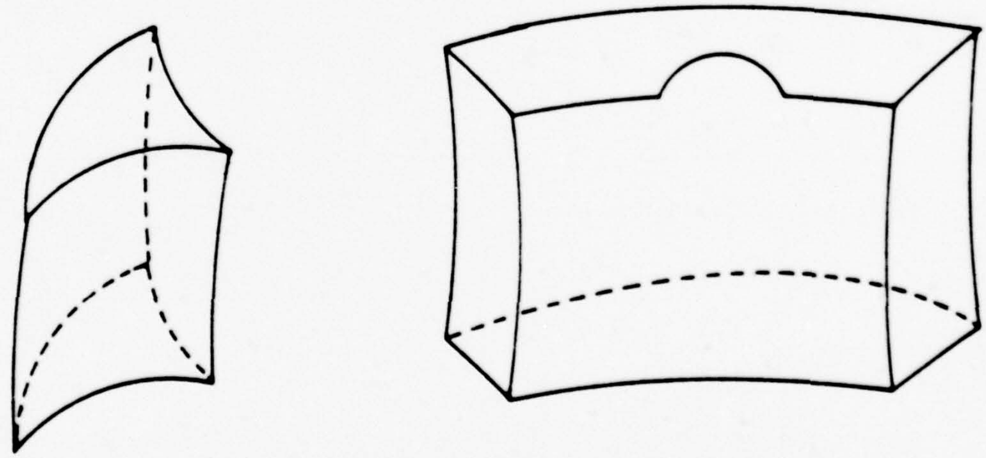
Captions to Figures



CURVES FROM KEY POINTS

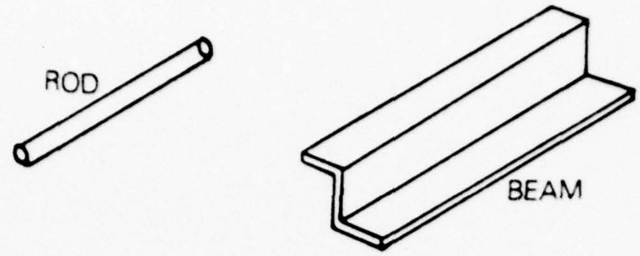


SURFACE PATCHES FROM CURVES



SOLID CHUNKS FROM SURFACE PATCHES

TRUSSES AND FRAMES



BEAM CROSS SECTIONAL AREAS

MEMBERANE AND AXISYMMETRIC ELEMENTS

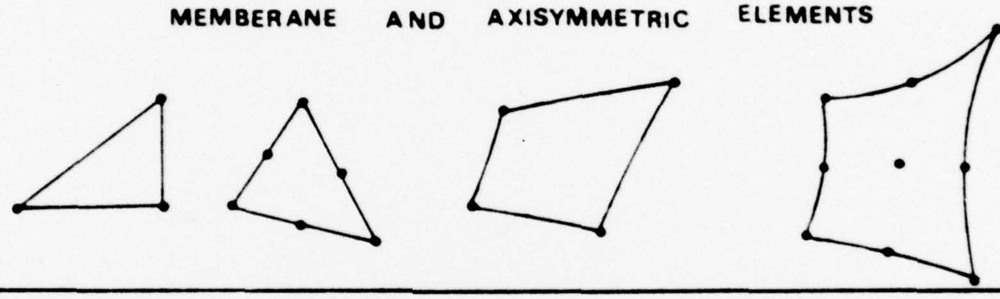
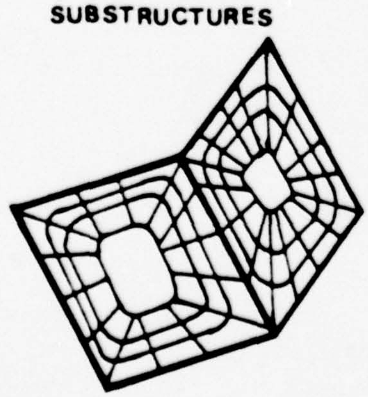
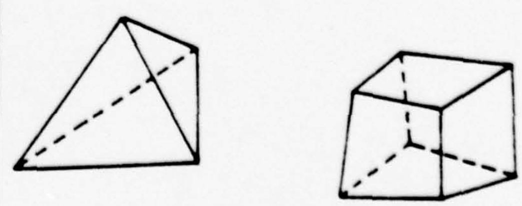


PLATE AND SHELL ELEMENTS

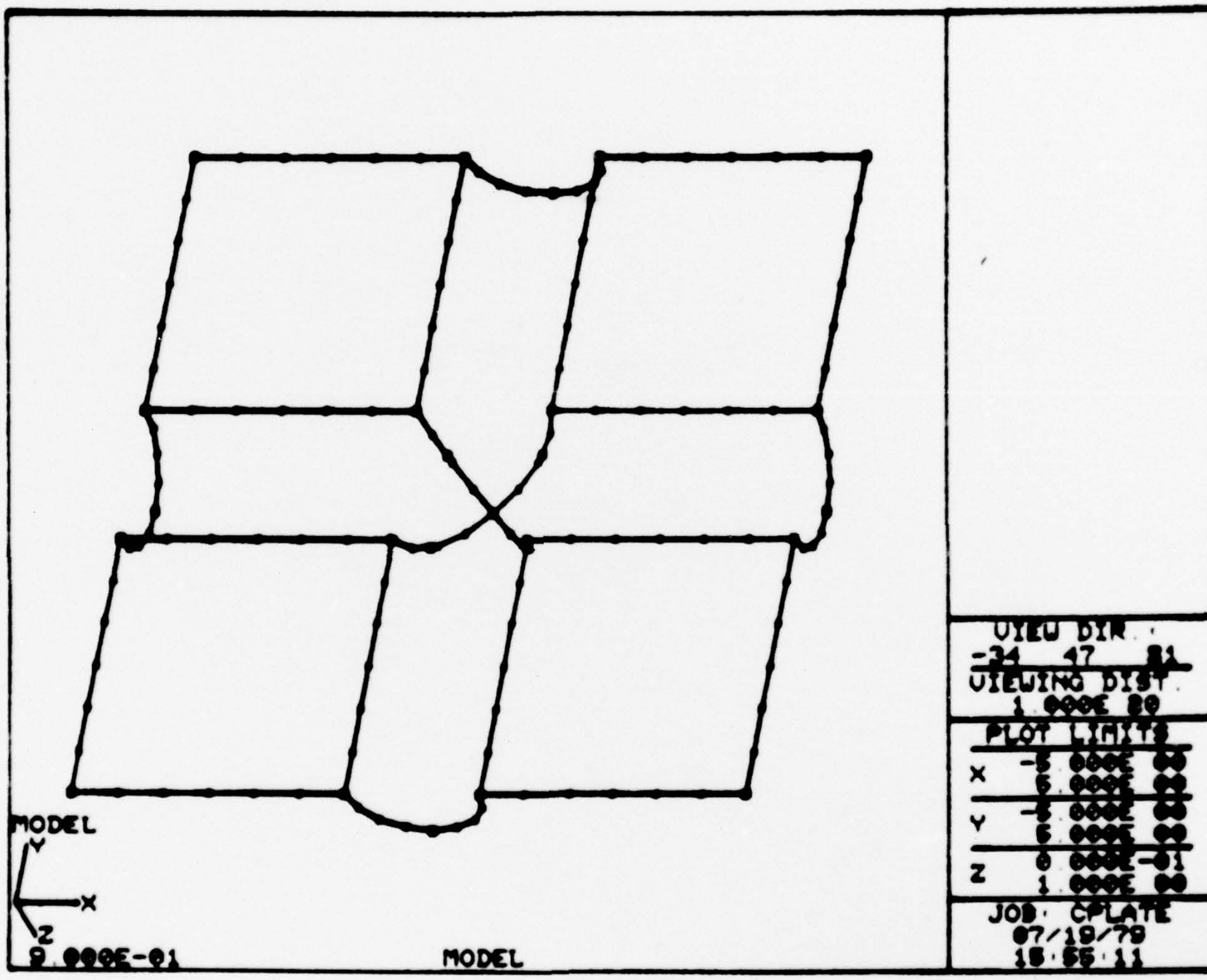
SUBSTRUCTURES

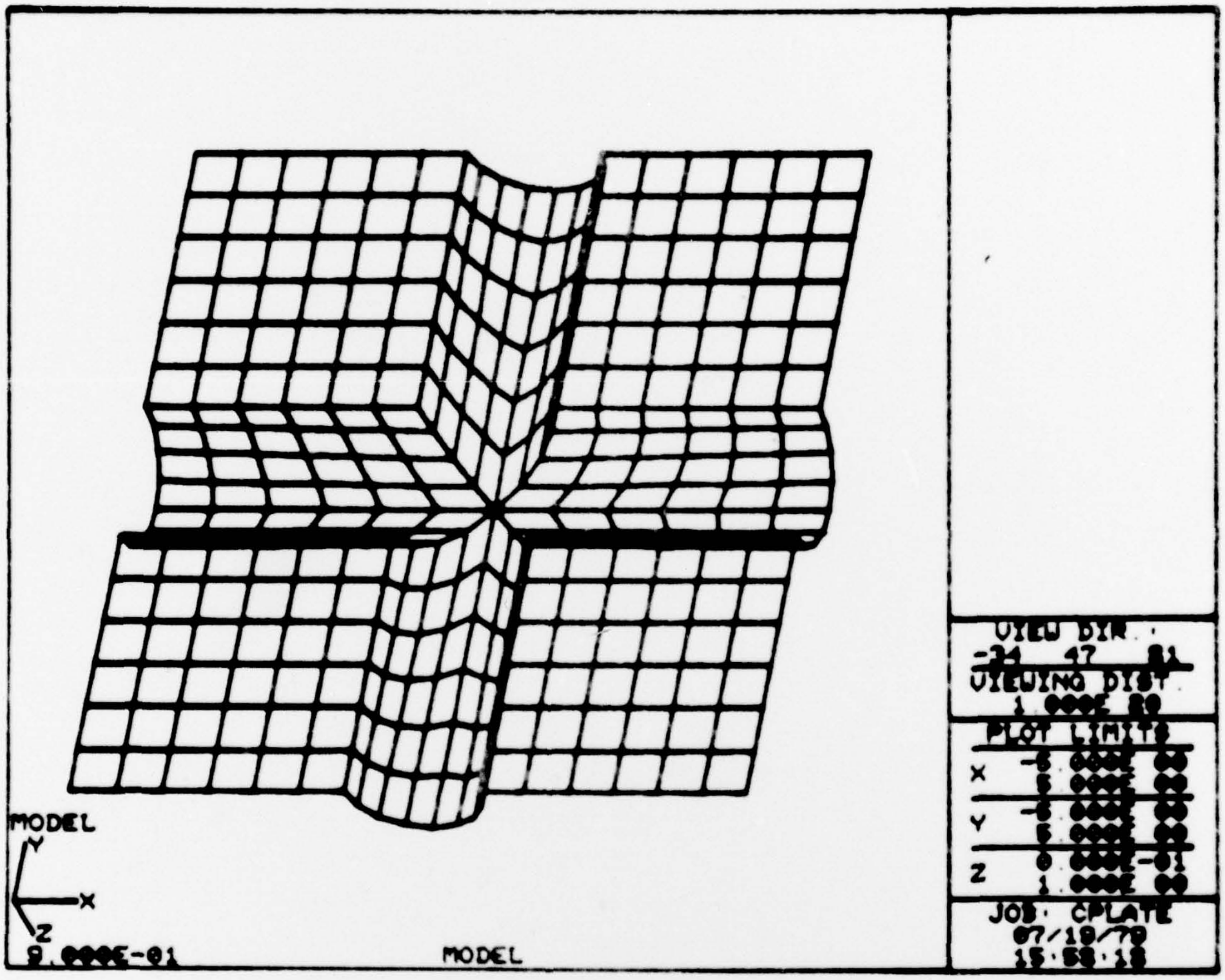


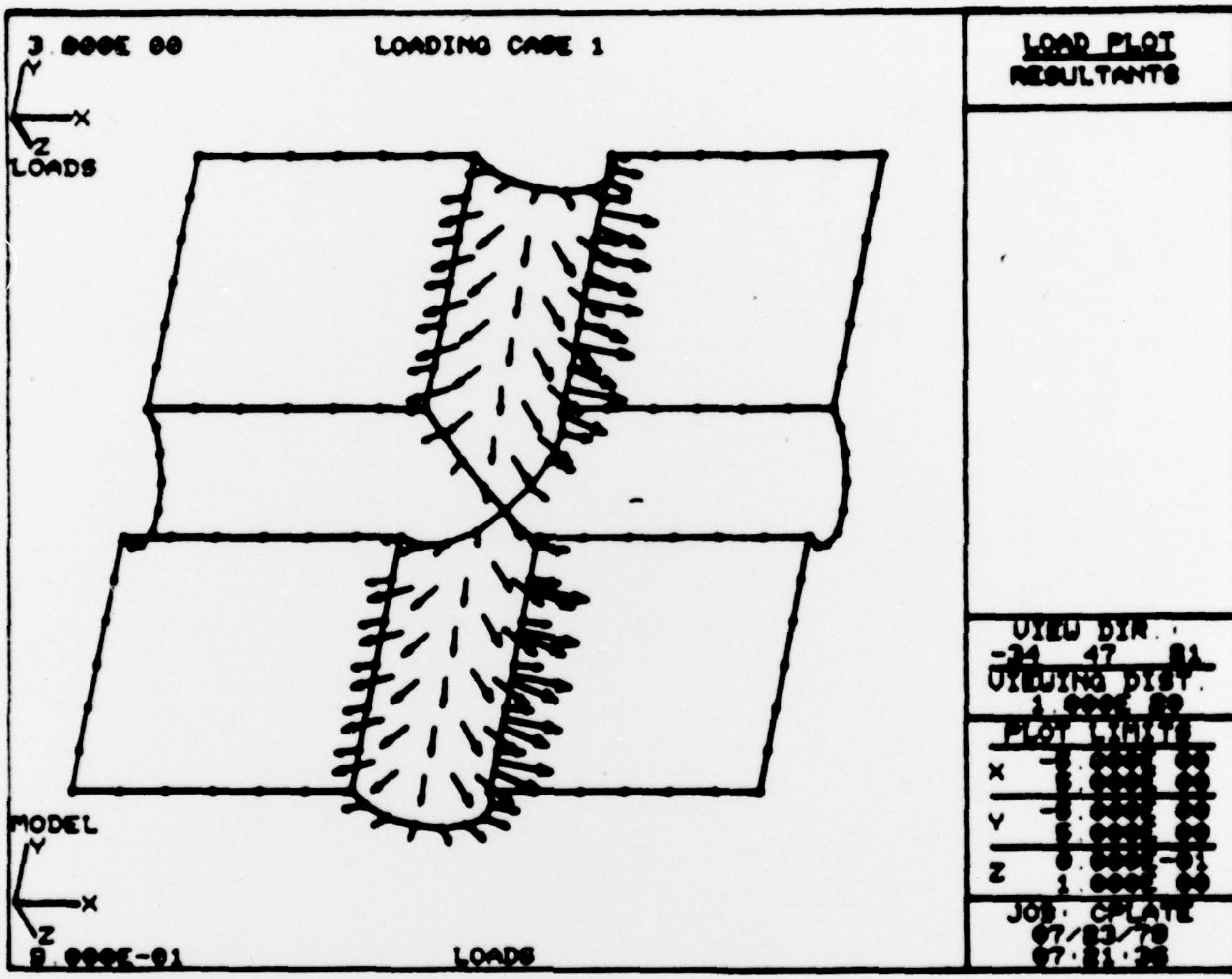
SOLID ELEMENTS
(Preprocessing only)



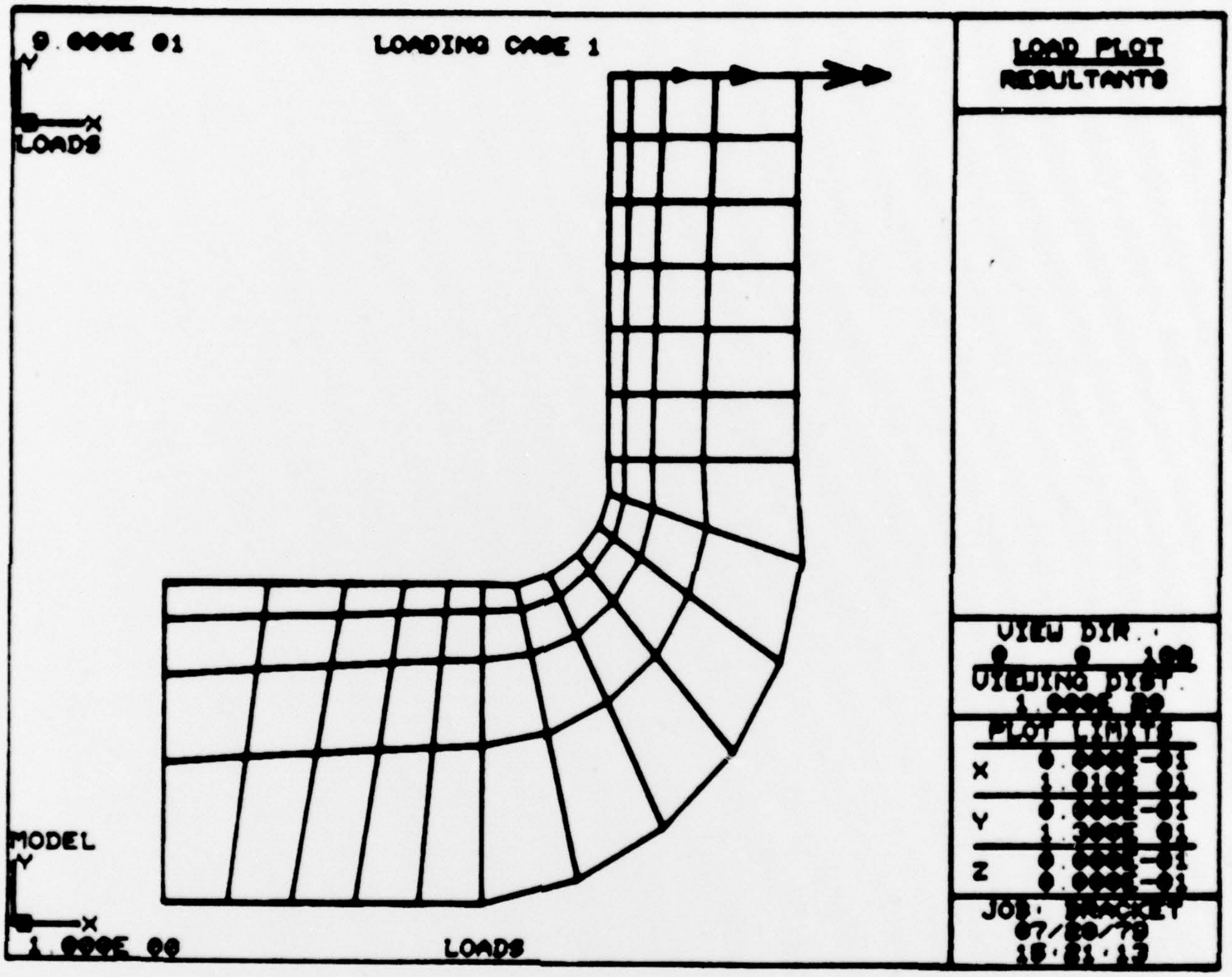
```
KPOINT/1/5,1/2/1,1/3/1,5/4/-1,5/5/-5,5/6/-1,1/7/-5,1
8/-5,-1/9/-1,-1/10/-5,-5/11/-1,-5/12/1,-5/13/1,-1
14/5,-5/15/5,-1/16/5,5/17/.70711,.70711,.70711
18/.70711,-.70711,.70711/19/-.70711,-.70711,.70711
20/-.70711,.70711,.70711/21/,5,1/22/5,,1/23/-5,,1
24/,-5,1/25/,,1/0
SLINE/L45/4,5,7/L316/3,16,7/L67/6,7,7/L12/1,2,7
L89/8,9,7/L1315/13,15,7/L1011/10,11,7/L1214/12,14,7
L57/5,7,7/L810/8,10,7/L46/4,6,7/L911/9,11,7
L23/2,3,7/L1213/12,13,7/L116/1,16,7/L1415/14,15,7/ /
CARC/C34/3,21,4,9/C1112/11,24,12,9/C78/7,23,8,9
C115/1,22,15,9/C625/6,20,25,5/C225/2,17,25,5
C925/9,19,25,5/C1325/13,18,25,5/ /
COMPLINE/L213/C225,C1325/L26/C225,C625/L69/C625,C925
L913/C925,C1325/ /
MSTEEL/1/0/ETH,1/0.036/0
GETY/QB4/1,1
GRID4/TL/L67,L46,L45,L57/TR/L12,L116,L316,L23
BL/L1011,L911,L89,L810/BR/L1214,L1415,L1315,L1213
TOP/L23,C34,L46,L26/BOTTOM/C1112,L1213,L913,L911
LEFT/C78,L67,L69,L89/RIGHT/C115,L12,L213,L1315/ /
END
```


MODEL
Y
X
Z
9.000E-01
MODEL
VIEW DIR
-34 47 81
VIEWING DIST
1.000E 00
PLOT LIMITS
X -5.000E 00
5.000E 00
Y -5.000E 00
5.000E 00
Z 0.000E-01
1.000E 00
JOB CPLATE
07/19/79
15:55:11

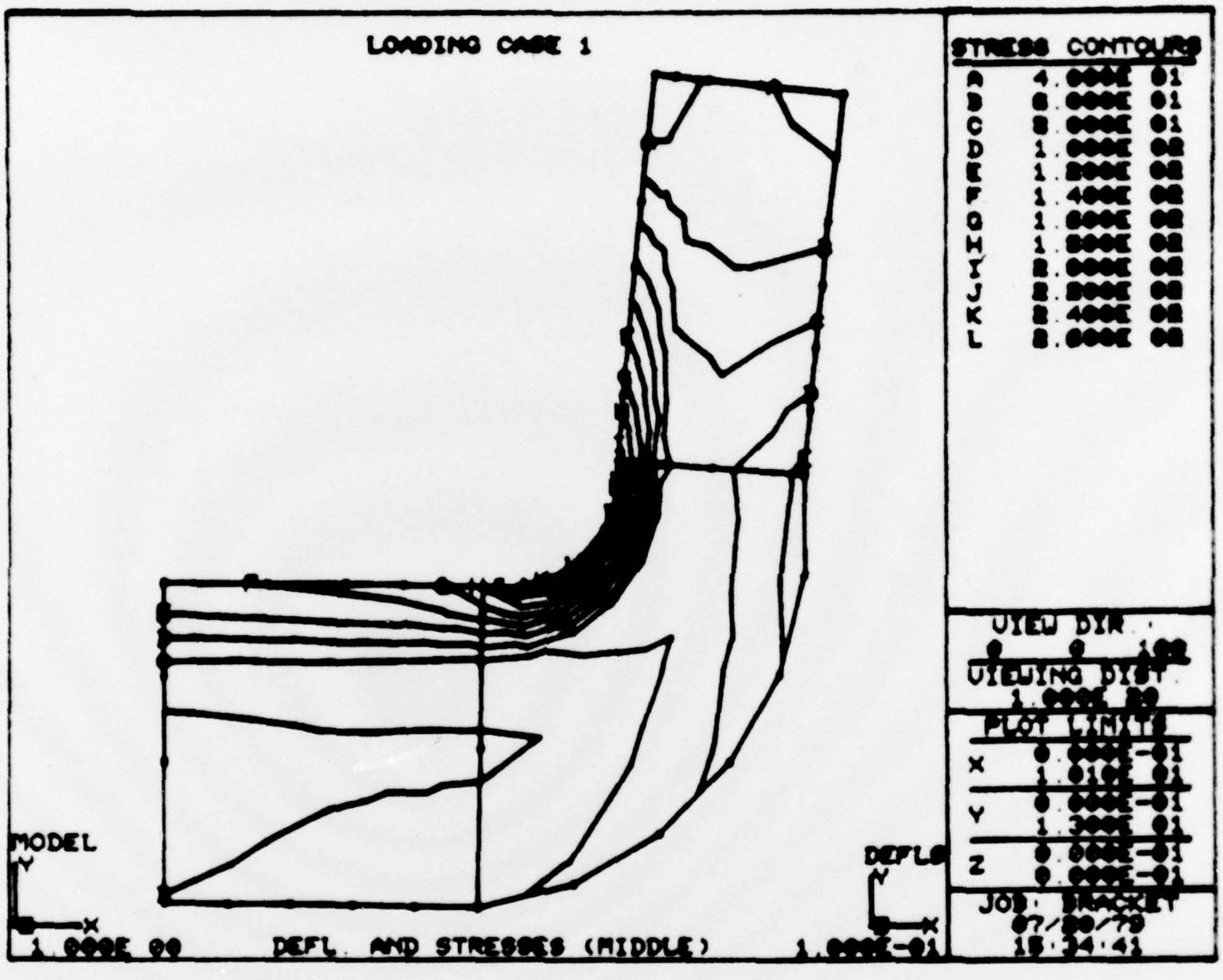

MODEL
Y
X
Z
MODEL
VIEW DIR
VIEWING DIST
PLOT LIMITS
X
Y
Z
JOB CPLATE

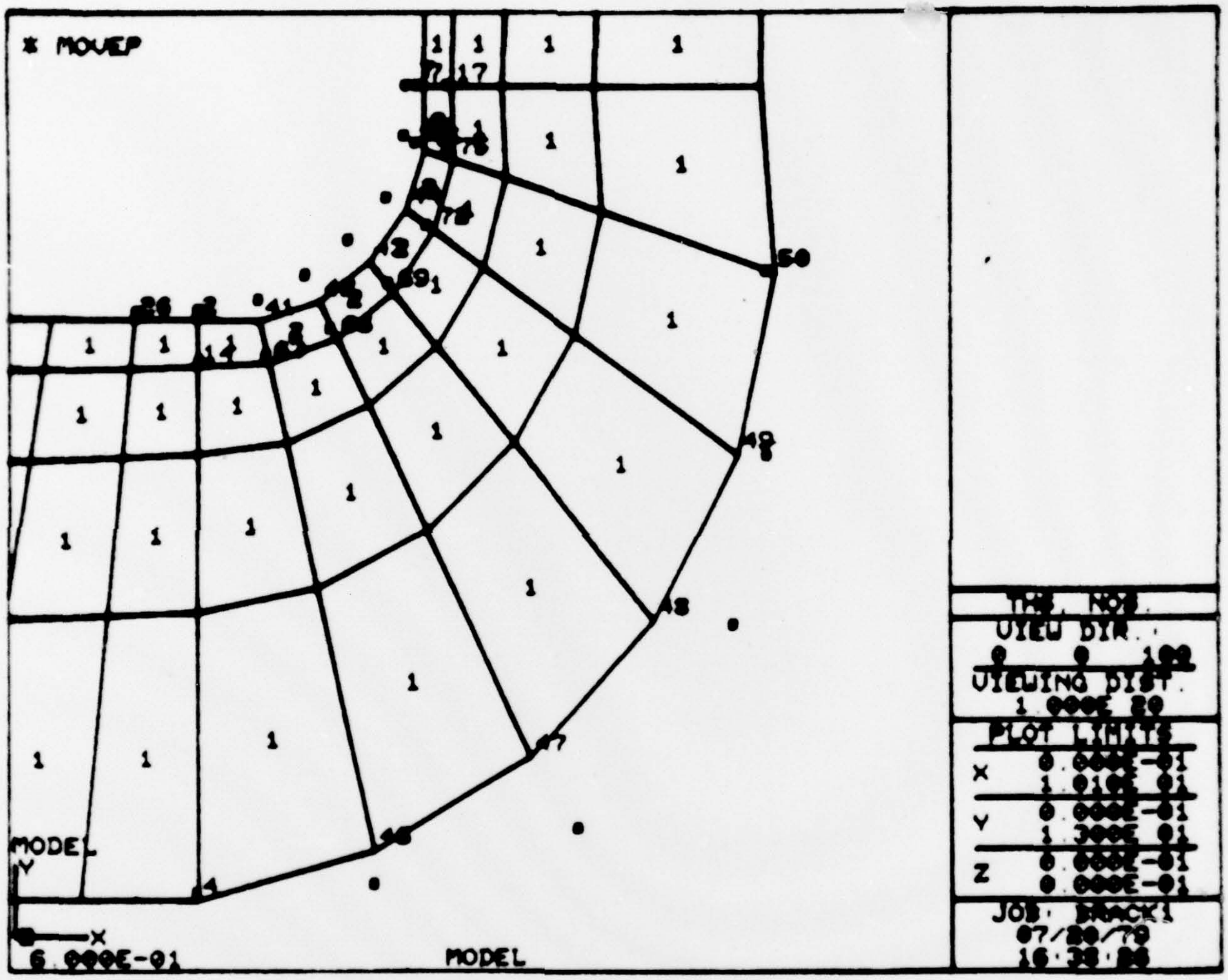


LOADING CASE 1
LOAD PLOT
RESULTANTS
LOADS
MODEL
LOADS
VIEW DIR.
VIEWING DIST.
PLOT LIMITS
X
Y
Z
JOB CPLATE


9.000E 01
LOADING CASE 1
LOADS
LOAD PLOT
RESULTANTS
VIEW DIR.
VIEWING DIST.
PLOT LIMITS
X
Y
Z
JOB:
MODEL
LOADS

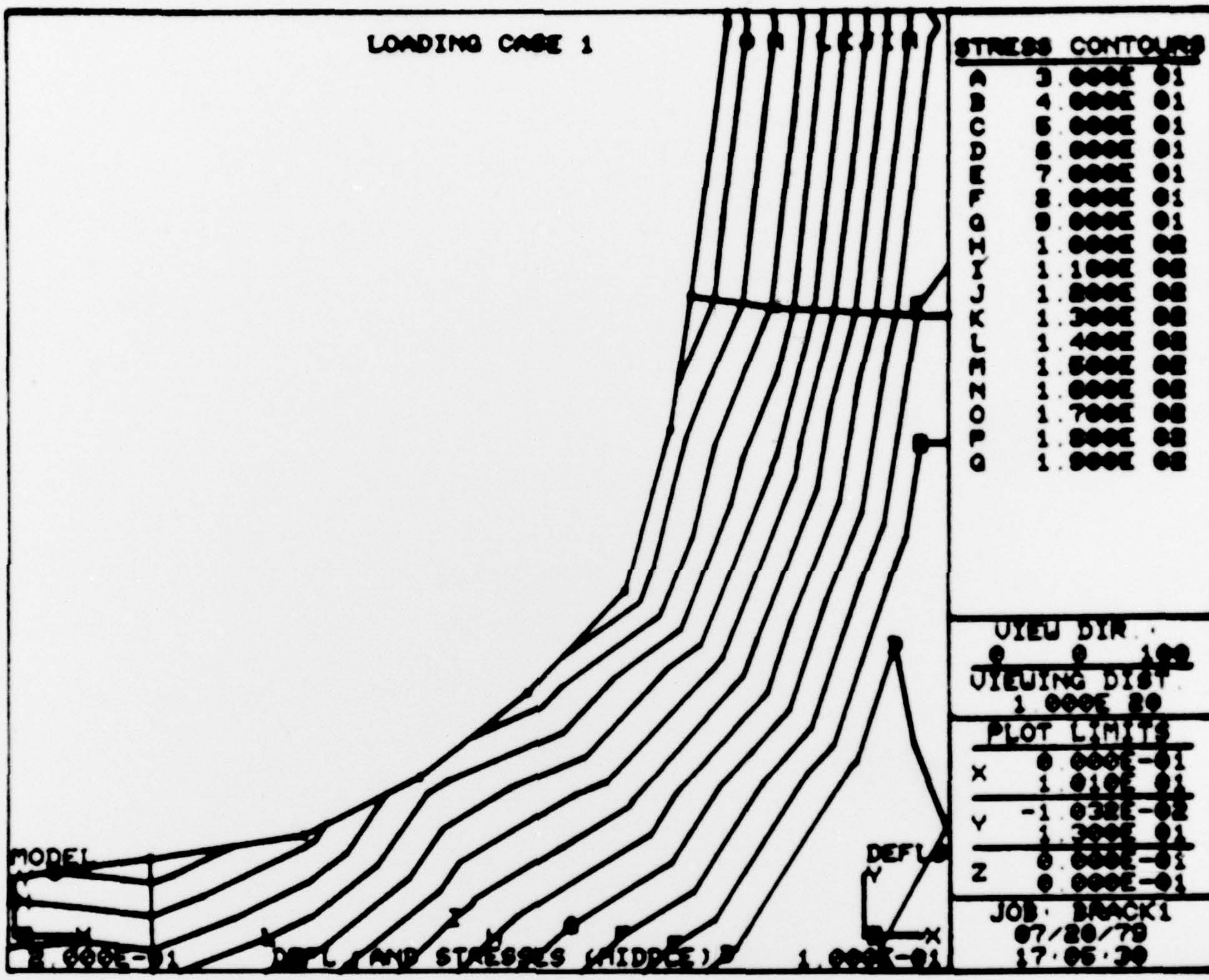
7


LOADING CASE 1
STRESS CONTOURS
A
B
C
D
E
F
G
H
I
J
K
L
MODEL
DEFL
DEFL. AND STRESSES (MIDDLE)
VIEW DIR
VIEWING DIST
PLOT LIMITS
X
Y
Z
JOB: BRACKET


* MOVEP
MODEL
Y
X
6.000E-01
MODEL
VIEW DIR.
VIEWING DIST.
PLOT LIMITS
X
Y
Z
JOB

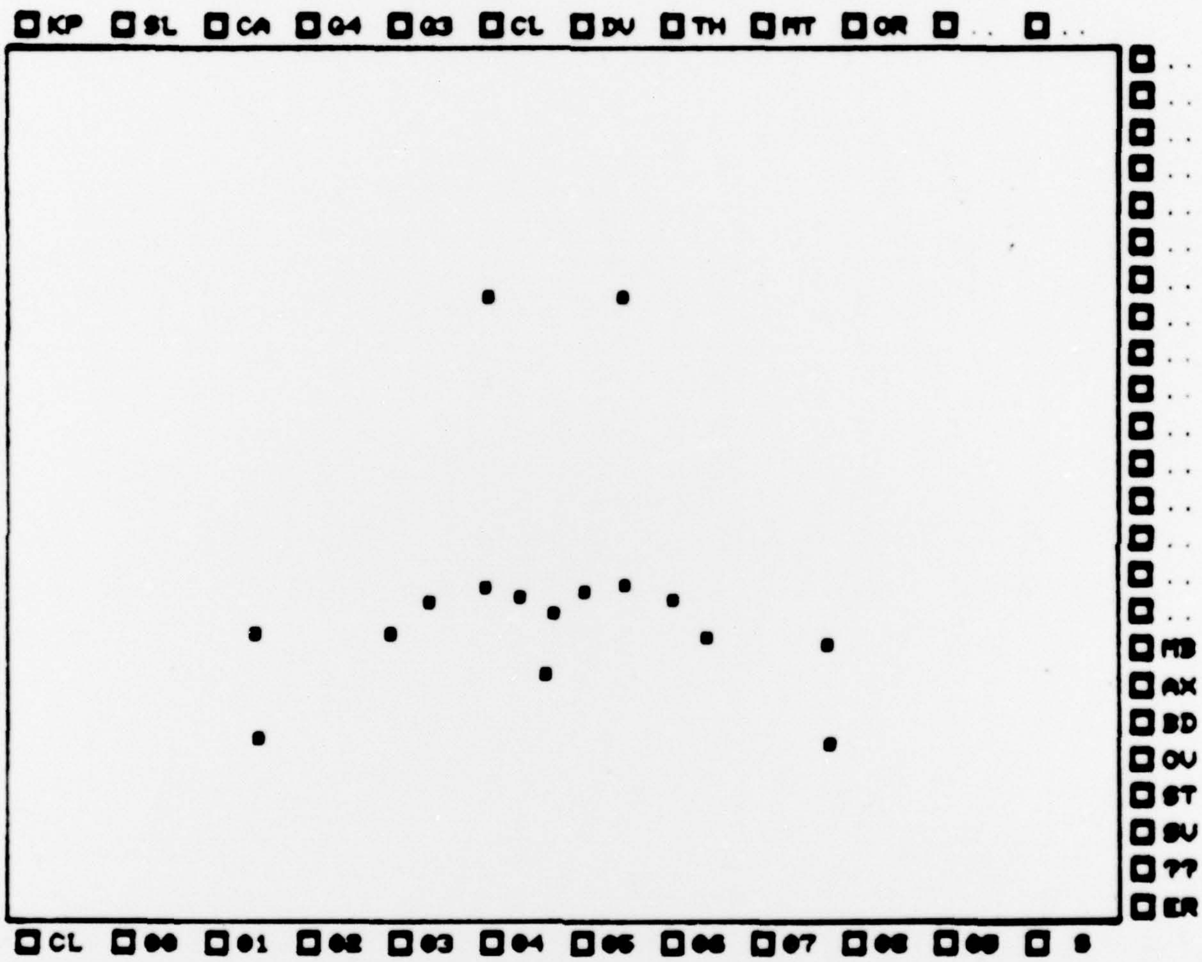
9

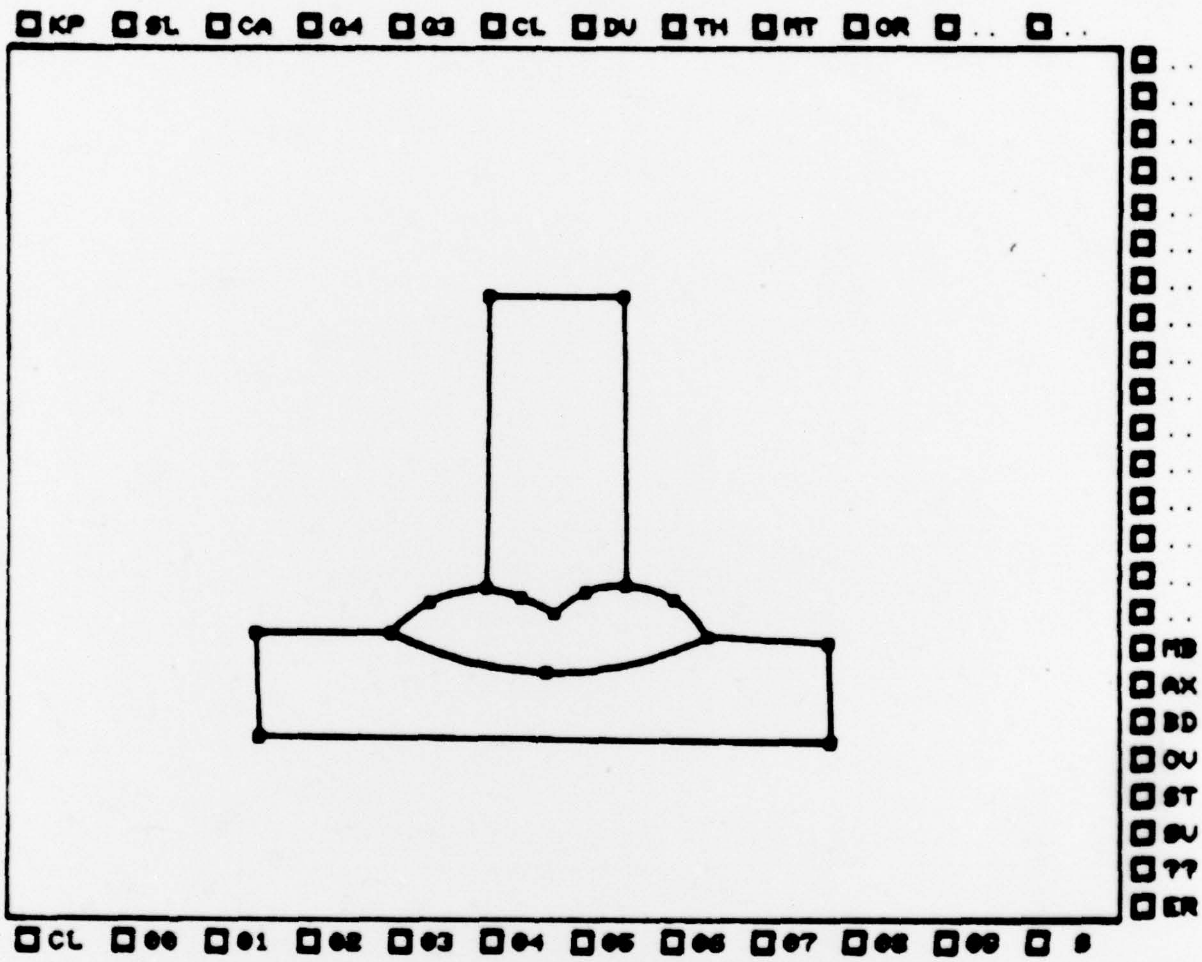

VIEW DIR.
VIEWING DIST.
PLOT LIMITS
X
Y
Z
JOB: BRACK1
07/20/79
MODEL
6.000E-01
MODEL



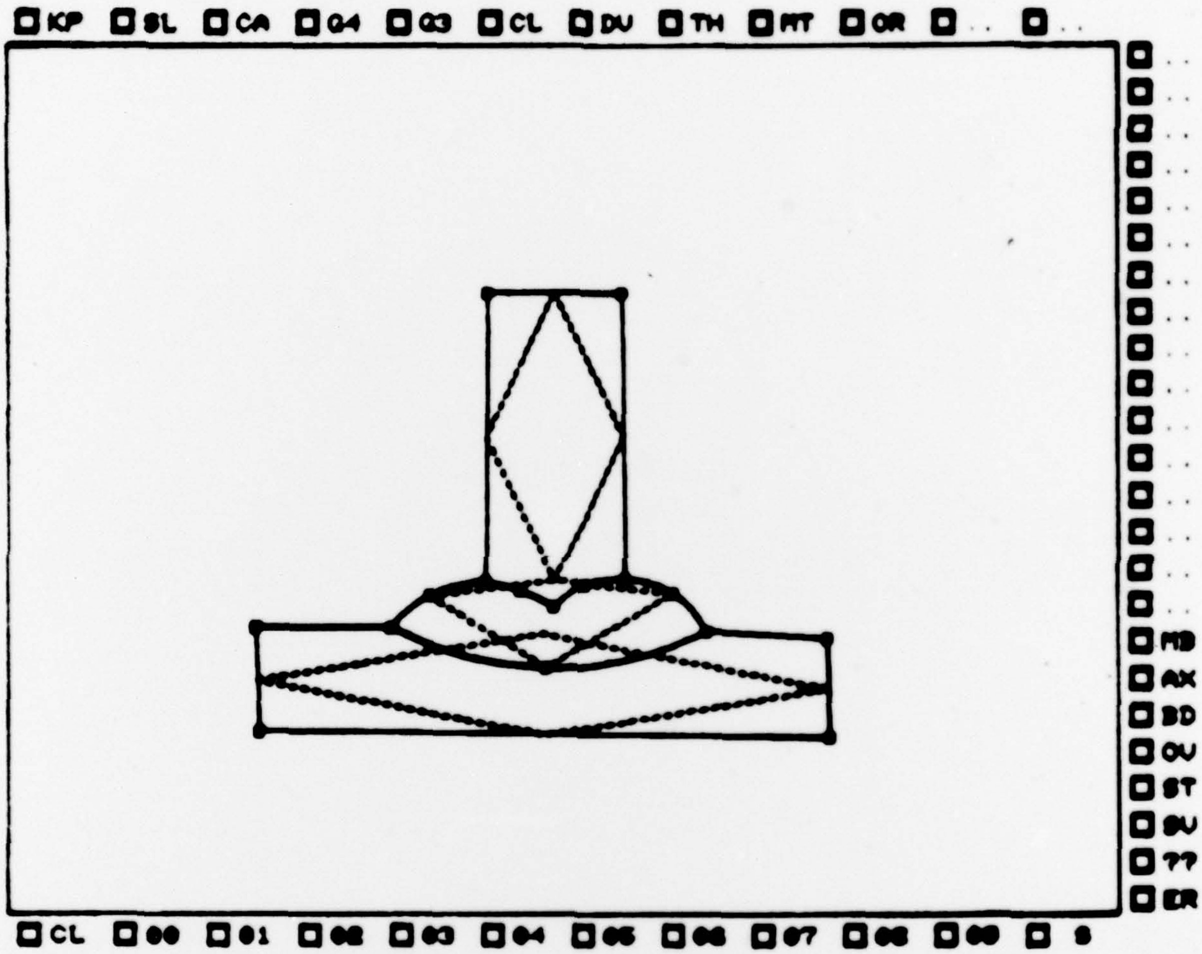
THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

11


KP
SL
CA
Q4
Q3
CL
DV
TH
MT
OR
MB
AX
BD
OV
ST
SU
??
ER
CL
01
03
04
07



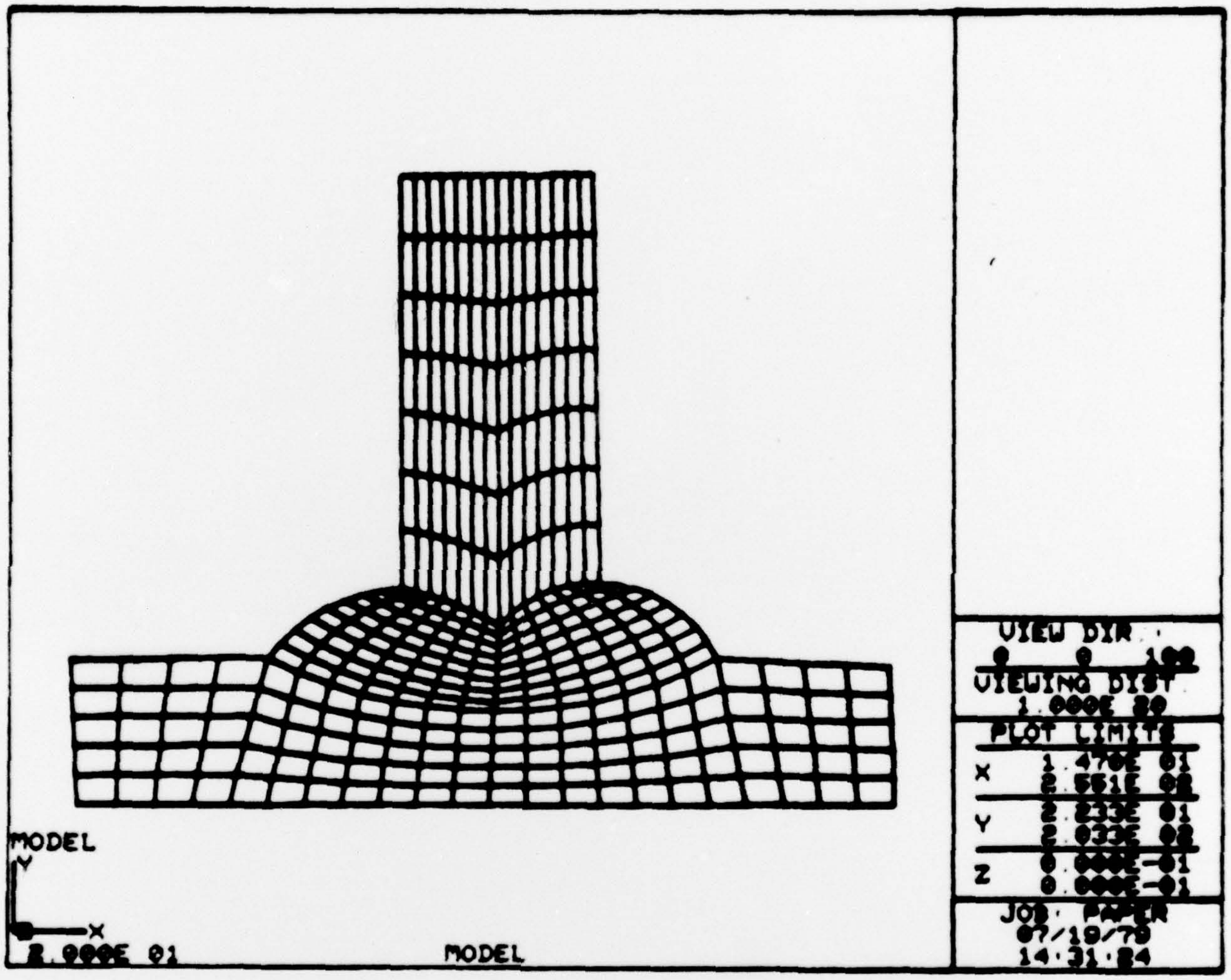
13

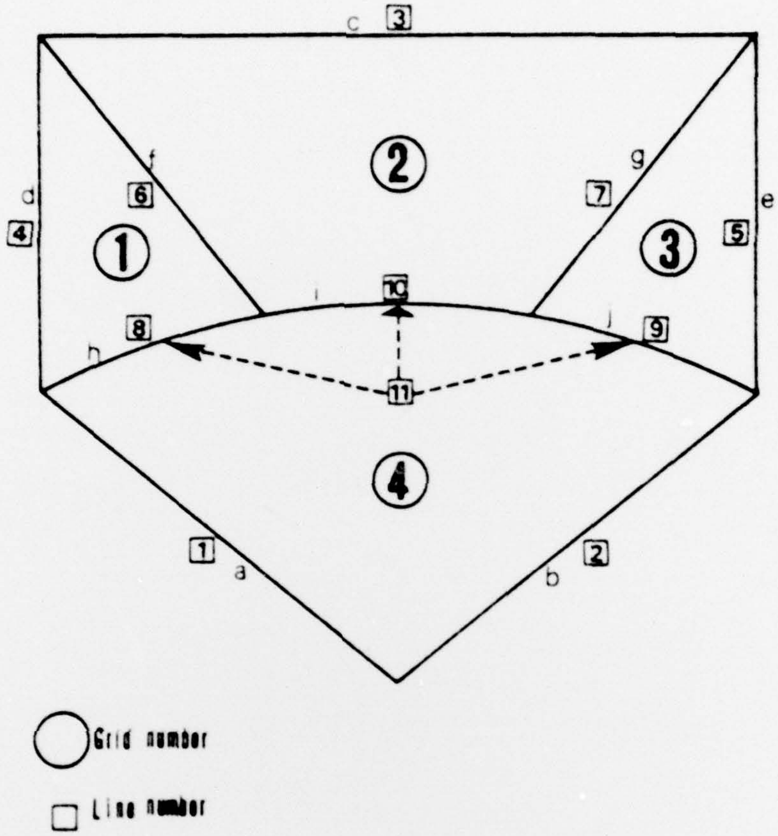


14

```
ETH,  3  /  1  /  .12000E+05  .27700E-01  .33567E+02

ELMAT,  4  /  1  /  .12000E+01  .95500E+01  .76201E+05  .44100E-01

KPOINT
    1      /  .16745850E+02,  .22331350E+02
    2      /  .14701060E+02,  .64209320E+02
    3      /  .71063330E+02,  .64947130E+02
    4      /  .86854620E+02,  .78124370E+02
    5      /  .11021910E+03,  .84416610E+02
    6      /  .12473600E+03,  .80615790E+02
    7      /  .13878660E+03,  .74314420E+02
    8      /  .15163750E+03,  .83462230E+02
    9      /  .16856340E+03,  .86178110E+02
   10      /  .18909200E+03,  .80460500E+02
   11      /  .20326020E+03,  .65180990E+02
   12      /  .13561520E+03,  .49828440E+02
   13      /  .10966920E+03,  .20264110E+03
   14      /  .16603150E+03,  .20337890E+03
   15      /  .25367640E+03,  .62847790E+02
   16      /  .25519620E+03,  .22958350E+02

SLINE
 SL0001      /   2,  1,   6
 SL0002      /   1, 16,  23
 SL0003      /  16, 15,   6
 SL0004      /  15, 11,   5
 SL0005      /   3,  2,   5
 SL0006      /  13,  5,   8
 SL0007      /  13, 14,  15
 SL0008      /  14,  9,   8

CARC
 CA0009      /   3,  4,  5,   9
 CA0010      /   5,  6,  7,   8
 CA0011      /   7,  8,  9,   8
 CA0012      /   9, 10, 11,   9
 CA0013      /  11, 12,  3,  15

COMPLINE
 CL0014      / SL0005, CA0013, SL0004
 CL0015      / CA0010, CA0011

GETY /QB4    /  1,  1  / GRID4  / G40001  / SL0001,CL0014,SL0003,SL0002

GETY /QB4    /  1,  1  / GRID4  / G40002  / CA0009,CL0015,CA0012,CA0013

GETY /QB4    /  1,  1  / GRID4  / G40003  / SL0007,SL0008,CL0015,SL0006

END
```

15


MODEL
Y
X
2.000E 01
MODEL
VIEW DIR
0 0 100
VIEWING DIST
1.000E 20
PLOT LIMITS
X 1.470E 01
2.551E 02
Y 2.233E 01
2.033E 02
Z 0.000E-01
0.000E-01
JOB PAPER
07/19/79
14:31:24


c 3
f
2
g
d
6
7
e
4
1
3
5
i
10
j
8
9
h
11
4
1
a
2
b
Grid number
Line number

TABLE(n X n)

	1	2	3	4	5	6	7	8	9	10	11
1		1									
2	1										
3										1	
4					1	1	1	1	1		
5				1		1	1	1	1		
6				1	1		1	1	1		
7				1	1	1		1	1		
8				1	1	1	1		1		2
9				1	1	1	1	1			2
10			1								2
11	1	1